


# 32장. String

## 32.1 String 생성자 함수

1. String 객체는 생성자 함수 객체다.  
따라서 new 연산자와 함께 호출하여 String 인스턴스를 생성할 수 있다.
2. ES5에서는 [[StringData]]를 [[PrimitiveValue]]라 불렀다.
3. 배열과 유사하게 인덱스를 사용하여 각 문자에 접근할 수 있다.
4. 단, 문자열은 원시 값이므로 변경할 수 없다. 이때 에러 발생 X
5. 생성자 함수 인수로 문자열이 아닌 값을 전달하면 인수를 문자열로 강제 변환한 후, [[StringData]] 내부 슬롯에 변환된 문자열을 할당한 String 래퍼 객체 생성
6. new 연산자 사용하지 않고 String 생성자 함수 호출하면 문자열을 반환한다.  
 이를 이용하여 명시적으로 타입을 변환하기도 함.

## 32.2 length 프로퍼티

1. 배열과 마찬가지로 length 프로퍼티를 갖는다.

## 32.3 String 메서드

1. 원본 객체를 직접 변경하는 메서드는 존재하지 않는다.  
즉, String 객체 메서드는 언제나 새로운 문자열을 반환.

### 32.3.1 String.prototype.indexOf

1. 대상 문자열에서 인수로 전달받은 문자열을 검색하여 첫 번째 인덱스를 반환.
  - 검색에 실패하면 -1 반환

2. ES6에서 도입된 `String.prototype.includes` 메서드를 사용하면 가독성이 더 좋다.

### 32.3.2 `String.prototype.search`

1. 인수로 전달받은 정규 표현식과 매치하는 문자열을 검색하여 일치하는 문자열의 인덱스를 반환
  - 검색에 실패하면 -1 반환

### 32.3.3 `String.prototype.includes`

1. 인수로 전달받은 문자열이 포함되어 있는지 확인하여 결과를 `boolean` 값으로 반환

### 32.3.4 `String.prototype.startsWith`

1. 대상 문자열이 인수로 전달받은 문자열로 시작하는지 확인하여 그 결과를 `boolean` 값으로 반환
2. 메서드의 2번째 인수로 검색을 시작할 인덱스를 전달할 수 있다.

### 32.3.5 `String.prototype.endsWith`

1. 대상 문자열이 인수로 전달받은 문자열로 끝나는지 확인하여 그 결과를 `boolean` 값으로 반환

### 32.3.6 `String.prototype.charAt`

1. 인수로 전달받은 인덱스에 위치한 문자를 검색하여 반환.
2. 인덱스가 문자열의 범위를 벗어난 정수인 경우 빈 문자열 반환.

### 32.3.7 `String.prototype.substring`

1. 첫 번째 인수 인덱스부터, 두 번째 인수 인덱스 바로 이전까지의 부분 문자열을 반환
  - 첫 번째 인수 > 두 번째 인수인 경우, 두 인수는 교환된다.
  - 인수 < 0 or NaN인 경우, 0으로 취급된다.
  - 인수 > 문자열 길이인 경우, 문자열 길이로 취급된다.

### 32.3.8 String.prototype.slice

1. substring 메서드와 동일하게 동작한다.  
단, slice 메서드에는 음수인 인수를 전달할 수 있다.

### 32.3.9 String.prototype.toUpperCase

1. 대상 문자열 → 모두 대문자

### 32.3.10 String.prototype.toLowerCase

1. 대상 문자열 → 모두 소문자

### 32.3.11 String.prototype.trim

1. 대상 문자열 앞뒤에 공백 문자가 있을 경우 이를 제거한 문자열 반환
  - trimStart : 앞 공백 제거
  - trimEnd : 뒤 공백 제거

### 32.3.12 String.prototype.repeat

1. 전달받은 정수만큼 반복해 연결한 새로운 문자열 반환
  - 전달받은 정수가 0이면 빈 문자열
  - 음수이면 RangeError 발생

### 32.3.13 String.prototype.replace

1. 첫 번째 인수로 전달받은 문자열 또는 정규표현식을 검색하여,  
두 번째 인수로 전달한 문자열로 치환한 문자열 반환
2. 검색된 문자열이 여럿 존재할 경우, 첫 번째로 검색된 문자열만 치환.

### 32.3.14 String.prototype.split

1. 첫 번째 인수로 전달한 문자열 or 정규 표현식을 검색하여 문자열을 구분한 후,  
분리된 각 문자열로 이루어진 배열을 반환.

2. 두 번째 인수로 배열의 길이를 지정할 수 있다.
3. `split` 메서드는 배열을 반환. 따라서 `reverse()`, `join()` 메서드와 함께 사용하여 문자열을 역순으로 뒤집을 수 있다.